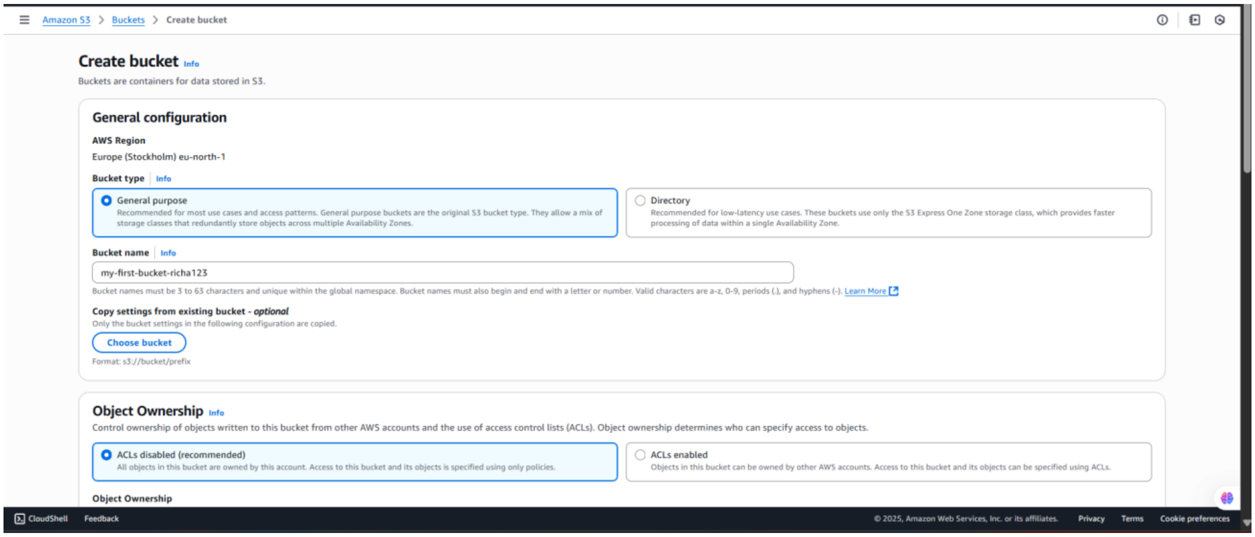
1.S3

Create a “bucket” to store file


Amazon S3 > Buckets > Create bucket
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Create bucket Info

Buckets are containers for data stored in S3.

General configuration

AWS Region

Europe (Stockholm) eu-north-1

Bucket type | Info

● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name | Info

my-first-bucket-richa123

Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). Learn More

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Choose bucket

Format: s3://bucket/prefix

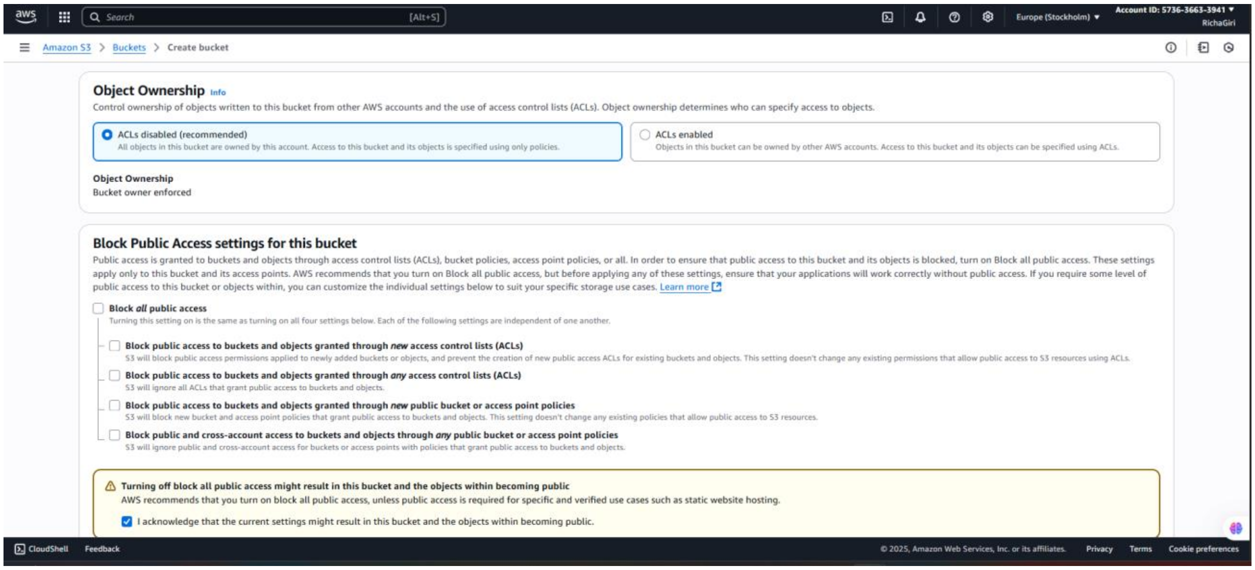
Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership


aws
Search [Alt+S]
Europe (Stockholm) ▾
Account ID: 5736-3663-3941 ▾
RichaGiri
Amazon S3 > Buckets > Create bucket
CloudShell
Feedback
© 2025, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Object Ownership Info

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

● ACLs disabled (recommended)
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

○ ACLs enabled
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

Object Ownership

Bucket owner enforced

Block Public Access settings for this bucket

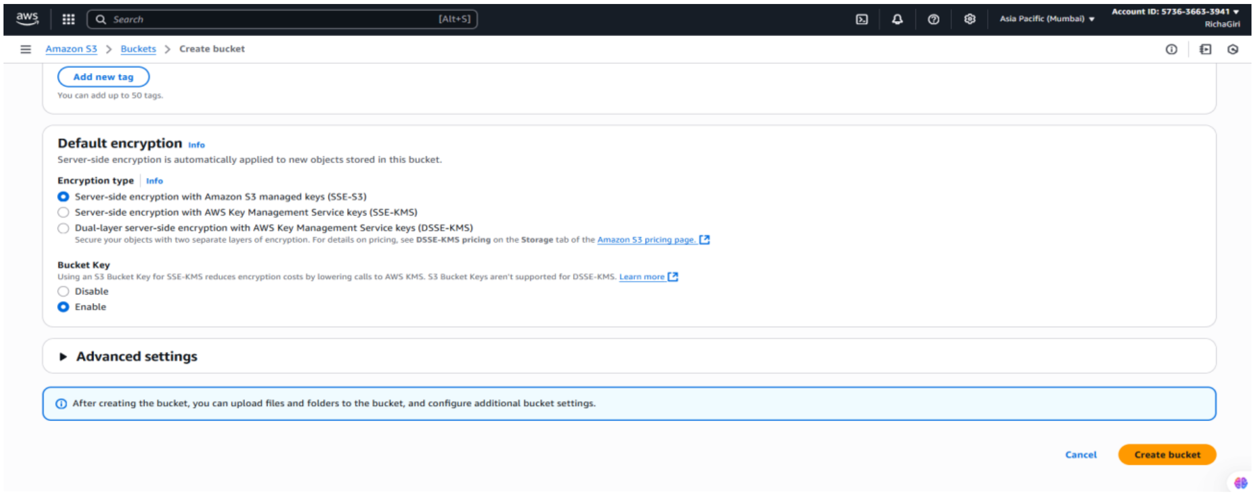
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more

□ **Block *all* public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

- □ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
  S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- □ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
  S3 will ignore all ACLs that grant public access to buckets and objects.
- □ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
  S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- □ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
  S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

⚠ **Turning off block all public access might result in this bucket and the objects within becoming public**
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☑ I acknowledge that the current settings might result in this bucket and the objects within becoming public.



You can add up to 50 tags.

Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type | Info

● Server-side encryption with Amazon S3 managed keys (SSE-S3)
○ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
○ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)
Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the Amazon S3 pricing page.

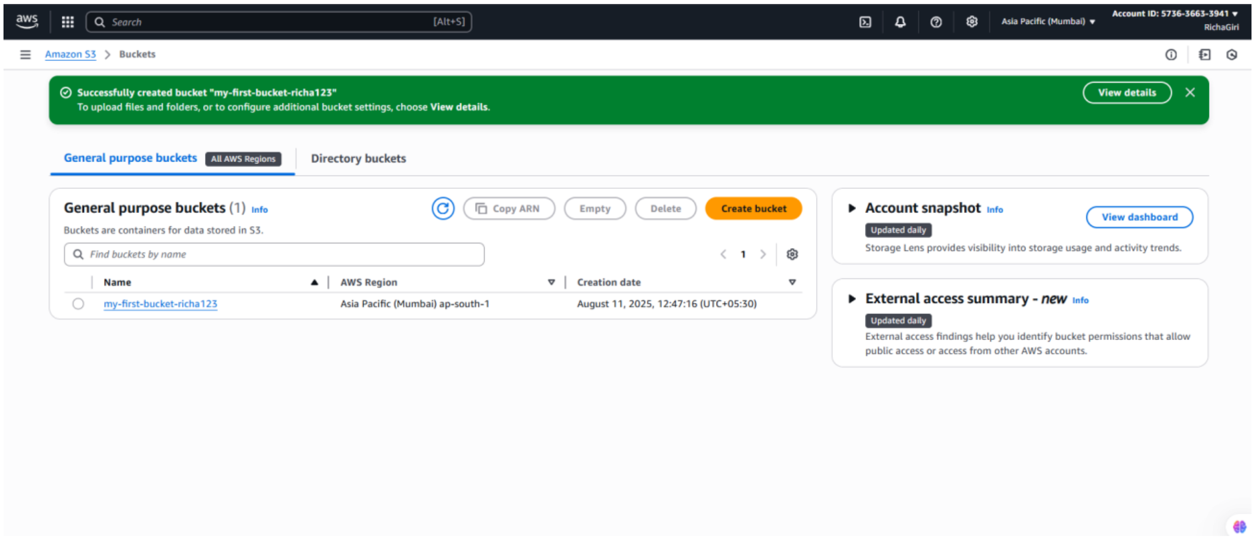
Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

○ Disable
● Enable

▶ **Advanced settings**

ⓘ After creating the bucket, you can upload files and folders to the bucket, and configure additional bucket settings.



Successfully created bucket "my-first-bucket-richa123"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

General purpose buckets (1) Info

Buckets are containers for data stored in S3.

Name	AWS Region	Creation date
○ my-first-bucket-richa123	Asia Pacific (Mumbai) ap-south-1	August 11, 2025, 12:47:16 (UTC+05:30)

Account snapshot Info

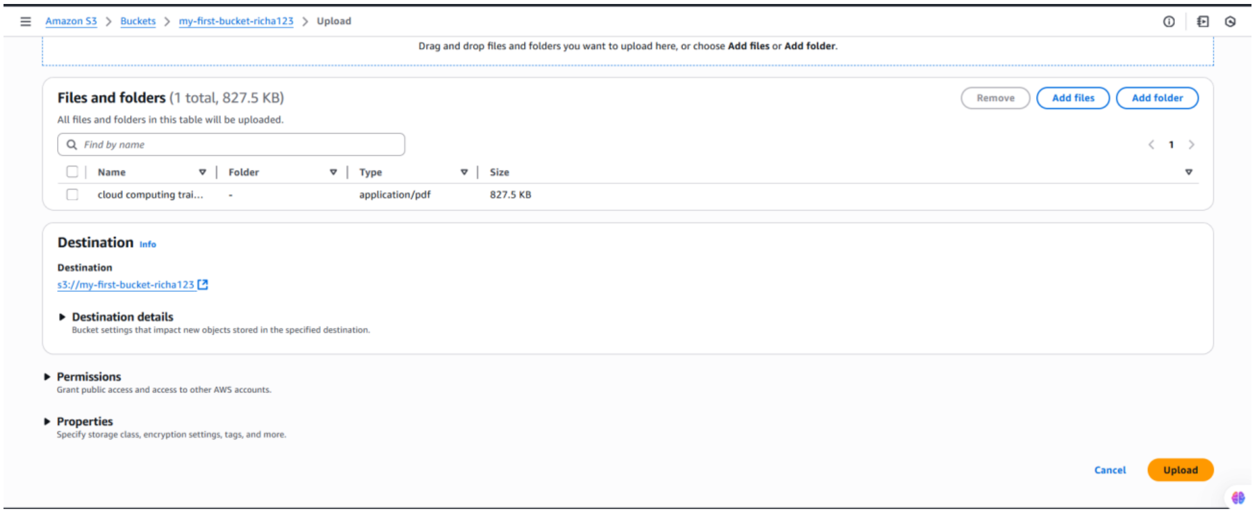
Updated daily

Storage Lens provides visibility into storage usage and activity trends.

External access summary - *new* Info

Updated daily

External access findings help you identify bucket permissions that allow public access or access from other AWS accounts.



Drag and drop files and folders you want to upload here, or choose **Add files** or **Add folder**.

Files and folders (1 total, 827.5 KB)

All files and folders in this table will be uploaded.

Name	Folder	Type	Size
□ cloud computing trai...	-	application/pdf	827.5 KB

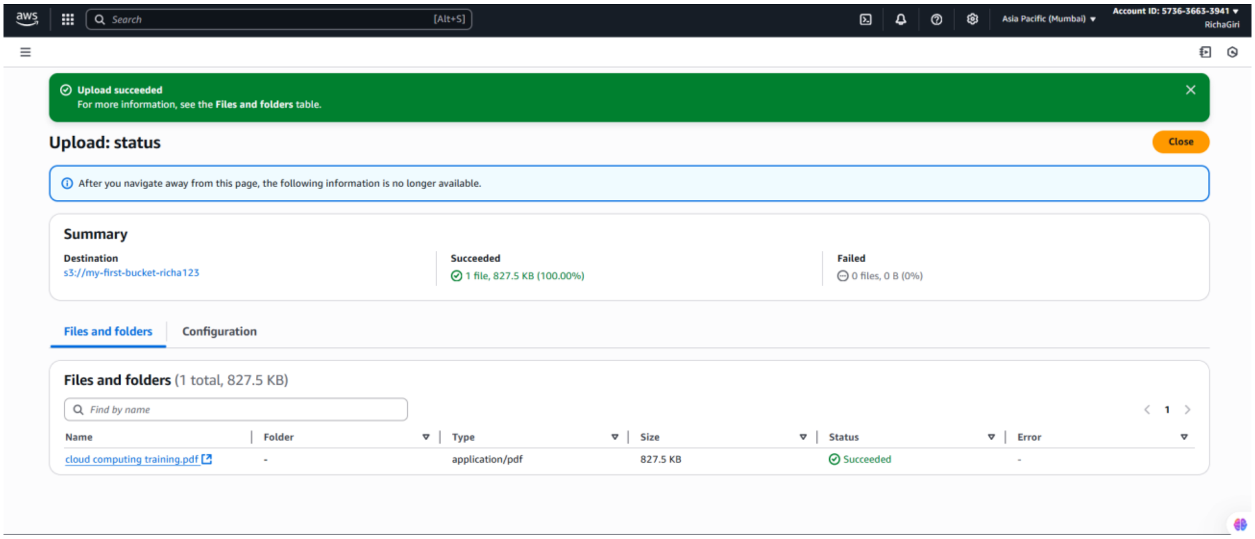
Destination Info

Destination
s3://my-first-bucket-richa123

▶ **Destination details**
Bucket settings that impact new objects stored in the specified destination.

▶ **Permissions**
Grant public access and access to other AWS accounts.

▶ **Properties**
Specify storage class, encryption settings, tags, and more.



Upload succeeded
For more information, see the **Files and folders** table.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

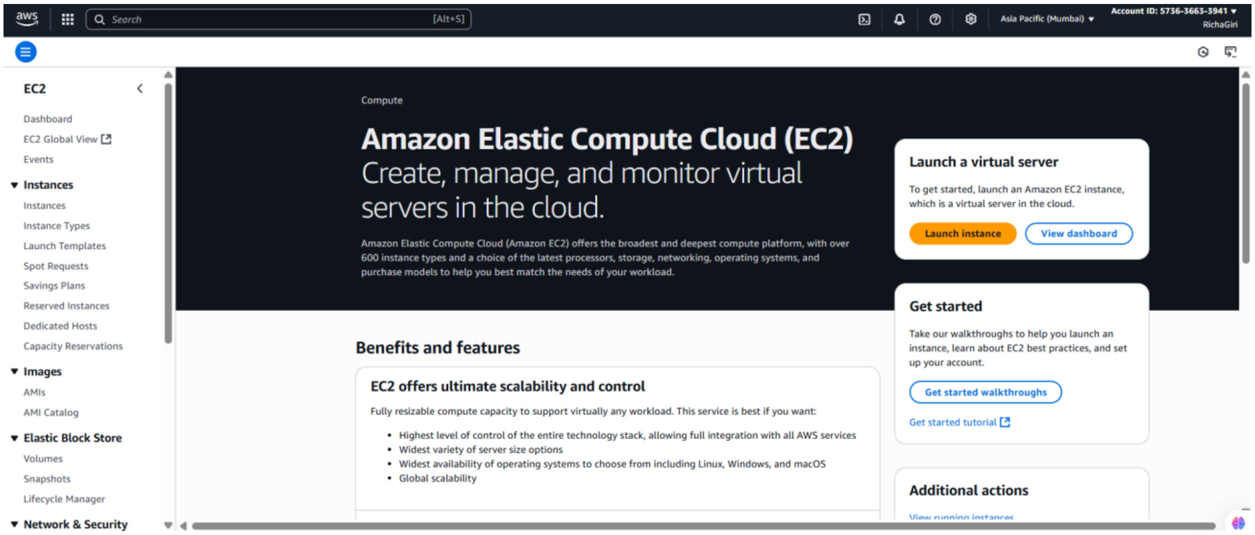
Destination	Succeeded	Failed
s3://my-first-bucket-richa123	✓ 1 file, 827.5 KB (100.00%)	0 files, 0 B (0%)

Files and folders (1 total, 827.5 KB)

Name	Folder	Type	Size	Status	Error
cloud computing training.pdf	-	application/pdf	827.5 KB	✓ Succeeded	-

2.EC2

Launch virtual machine(instance)



Compute

Amazon Elastic Compute Cloud (EC2)
Create, manage, and monitor virtual servers in the cloud.

Amazon Elastic Compute Cloud (Amazon EC2) offers the broadest and deepest compute platform, with over 600 instance types and a choice of the latest processors, storage, networking, operating systems, and purchase models to help you best match the needs of your workload.

Launch a virtual server

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Get started

Take our walkthroughs to help you launch an instance, learn about EC2 best practices, and set up your account.

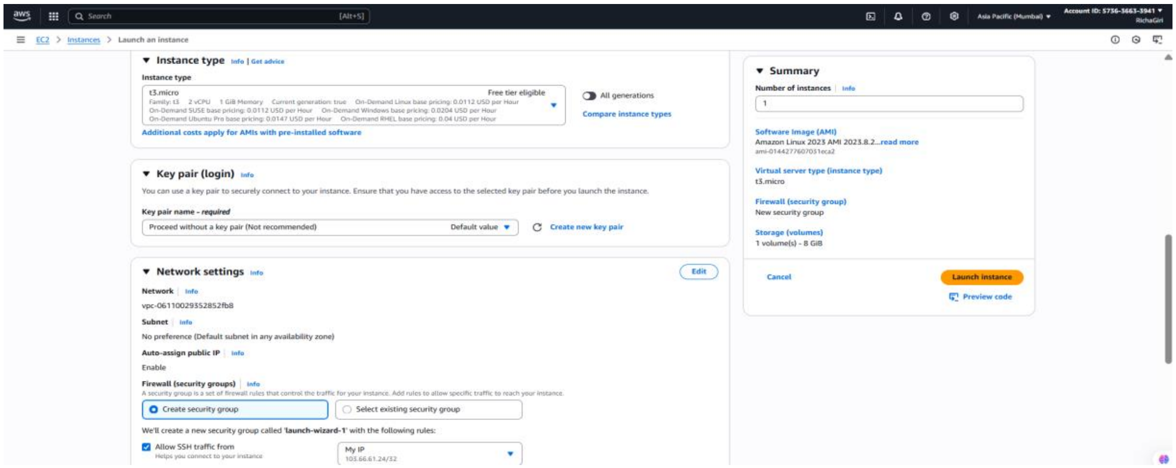
Additional actions

Benefits and features

EC2 offers ultimate scalability and control

Fully resizable compute capacity to support virtually any workload. This service is best if you want:

- Highest level of control of the entire technology stack, allowing full integration with all AWS services
- Widest variety of server size options
- Widest availability of operating systems to choose from including Linux, Windows, and macOS
- Global scalability



▾ Instance type

Instance type

t3.micro — Free tier eligible
Family: t3 2 vCPU 1 GiB Memory Current generation: true On-Demand Linux base pricing: 0.0112 USD per Hour
On-Demand SUSE base pricing: 0.0112 USD per Hour On-Demand Windows base pricing: 0.0204 USD per Hour
On-Demand Ubuntu Pro base pricing: 0.0147 USD per Hour On-Demand RHEL base pricing: 0.04 USD per Hour

All generations

Additional costs apply for AMIs with pre-installed software

▾ Key pair (login)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*: Proceed without a key pair (Not recommended) — Default value

▾ Network settings

Network: vpc-06110029352852fb8

Subnet: No preference (Default subnet in any availability zone)

Auto-assign public IP: Enable

Firewall (security groups)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

● Create security group ○ Select existing security group

We'll create a new security group called '**launch-wizard-1**' with the following rules:

☑ Allow SSH traffic from — My IP 103.66.61.24/32
Helps you connect to your instance

▾ Summary

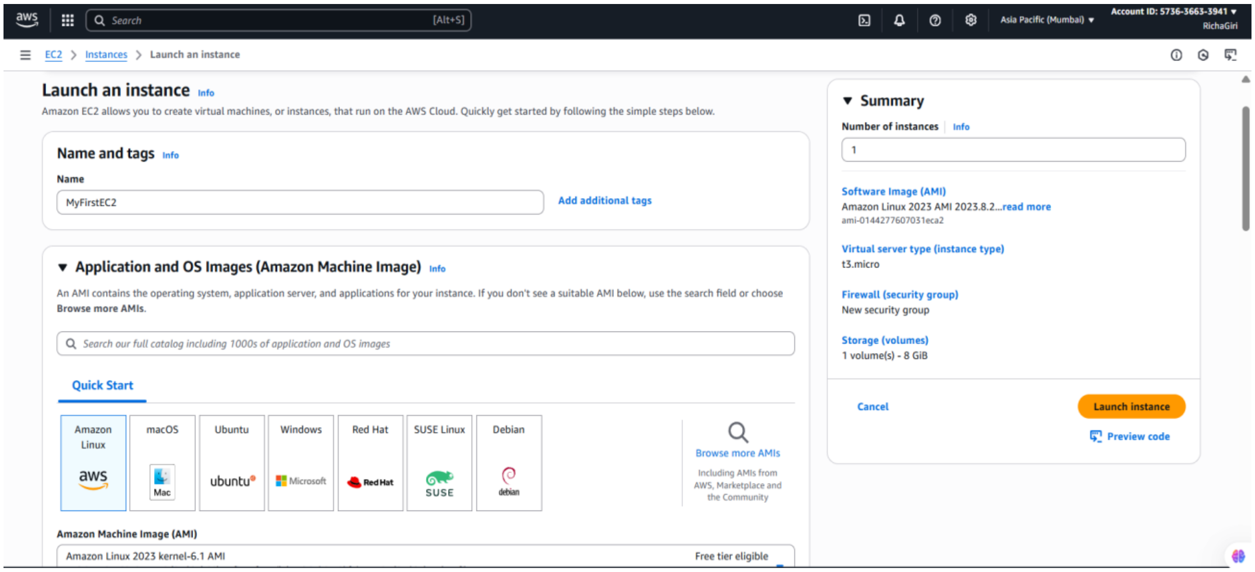
Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.8.2...read more
ami-0144277607031eca2

Virtual server type (instance type): t3.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB



Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags

Name: MyFirstEC2

▾ Application and OS Images (Amazon Machine Image)

An AMI contains the operating system, application server, and applications for your instance. If you don't see a suitable AMI below, use the search field or choose Browse more AMIs.

Search our full catalog including 1000s of application and OS images

Amazon Machine Image (AMI): Amazon Linux 2023 kernel-6.1 AMI — Free tier eligible

▾ Summary

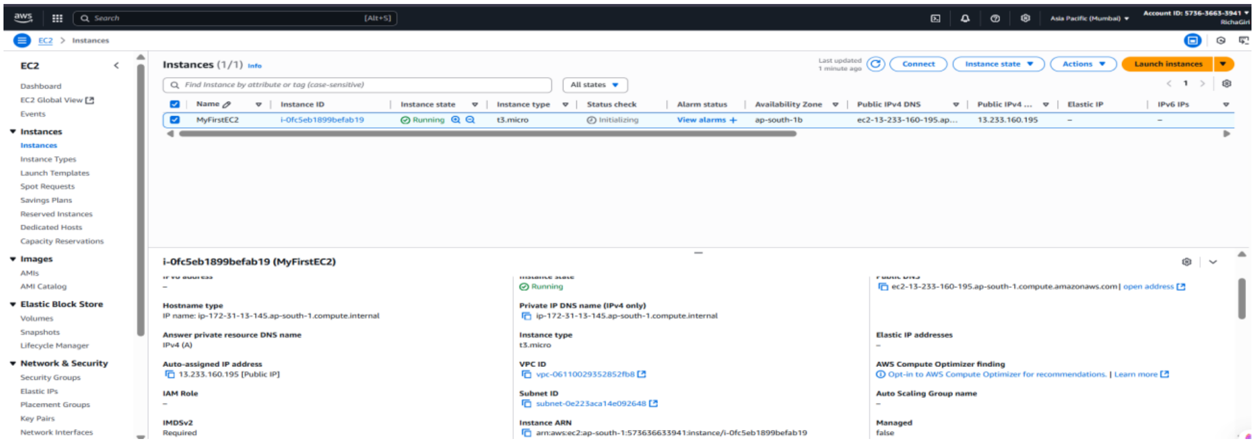
Number of instances: 1

Software Image (AMI): Amazon Linux 2023 AMI 2023.8.2...read more
ami-0144277607031eca2

Virtual server type (instance type): t3.micro

Firewall (security group): New security group

Storage (volumes): 1 volume(s) - 8 GiB



Instances (1/1) Info

Find Instance by attribute or tag (case-sensitive)

☑	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4 ...	Elastic IP	IPv6 IPs
☑	MyFirstEC2	i-0fc5eb1899befab19	Running	t3.micro	Initializing	View alarms +	ap-south-1b	ec2-13-233-160-195.ap...	13.233.160.195	–	–

i-0fc5eb1899befab19 (MyFirstEC2)

IPv6 address
–

Hostname type
IP name: ip-172-31-13-145.ap-south-1.compute.internal

Answer private resource DNS name
IPv4 (A)

Auto-assigned IP address
13.233.160.195 [Public IP]

IAM Role
–

IMDSv2
Required

Instance state
Running

Private IP DNS name (IPv4 only)
ip-172-31-13-145.ap-south-1.compute.internal

Instance type
t3.micro

VPC ID
vpc-06110029352852fb8

Subnet ID
subnet-0e223aca14e092648

Instance ARN
arn:aws:ec2:ap-south-1:573636633941:instance/i-0fc5eb1899befab19

Public DNS
ec2-13-233-160-195.ap-south-1.compute.amazonaws.com | open address

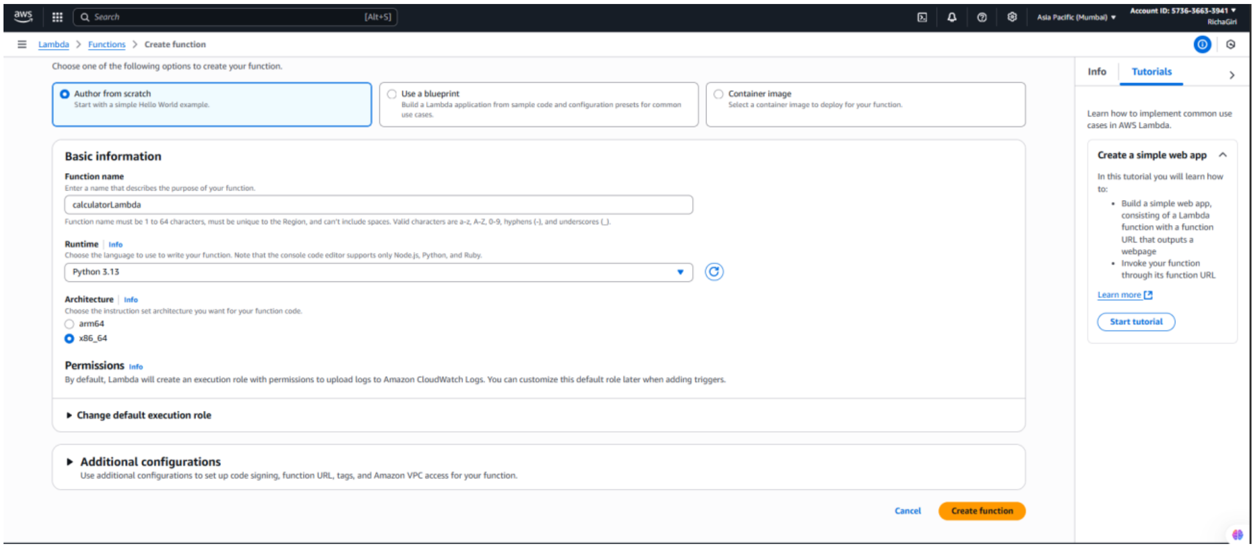
Elastic IP addresses
–

AWS Compute Optimizer finding
Opt-in to AWS Compute Optimizer for recommendations. | Learn more

Auto Scaling Group name
–

Managed
false

3.Lambda



Choose one of the following options to create your function.

● Author from scratch
Start with a simple Hello World example.

○ Use a blueprint
Build a Lambda application from sample code and configuration presets for common use cases.

○ Container image
Select a container image to deploy for your function.

Basic information

Function name
Enter a name that describes the purpose of your function.
calculatorLambda
Function name must be 1 to 64 characters, must be unique to the Region, and can't include spaces. Valid characters are a-z, A-Z, 0-9, hyphens (-), and underscores (_).

Runtime | Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Python 3.13

Architecture | Info
Choose the instruction set architecture you want for your function code.
○ arm64
● x86_64

Permissions Info
By default, Lambda will create an execution role with permissions to upload logs to Amazon CloudWatch Logs. You can customize this default role later when adding triggers.

▸ Change default execution role

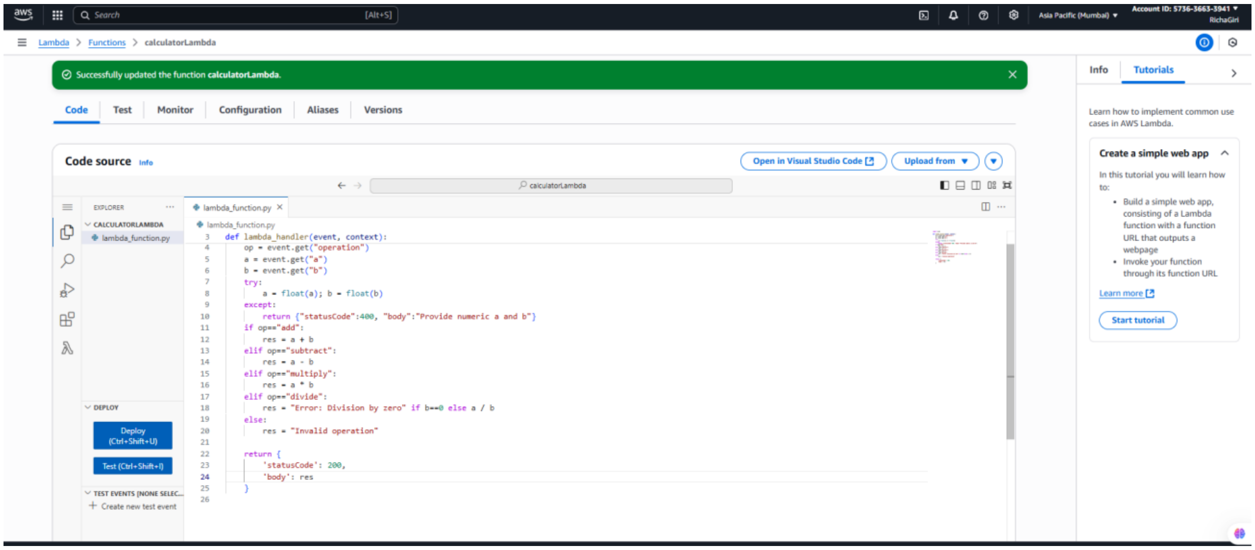
▸ Additional configurations
Use additional configurations to set up code signing, function URL, tags, and Amazon VPC access for your function.

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL



Successfully updated the function **calculatorLambda**.

Code source

```python
def lambda_handler(event, context):
    op = event.get("operation")
    a = event.get("a")
    b = event.get("b")
    try:
        a = float(a); b = float(b)
    except:
        return {"statusCode":400, "body":"Provide numeric a and b"}
    if op=="add":
        res = a + b
    elif op=="subtract":
        res = a - b
    elif op=="multiply":
        res = a * b
    elif op=="divide":
        res = "Error: Division by zero" if b==0 else a / b
    else:
        res = "Invalid operation"

    return {
        'statusCode': 200,
        'body': res
    }
```

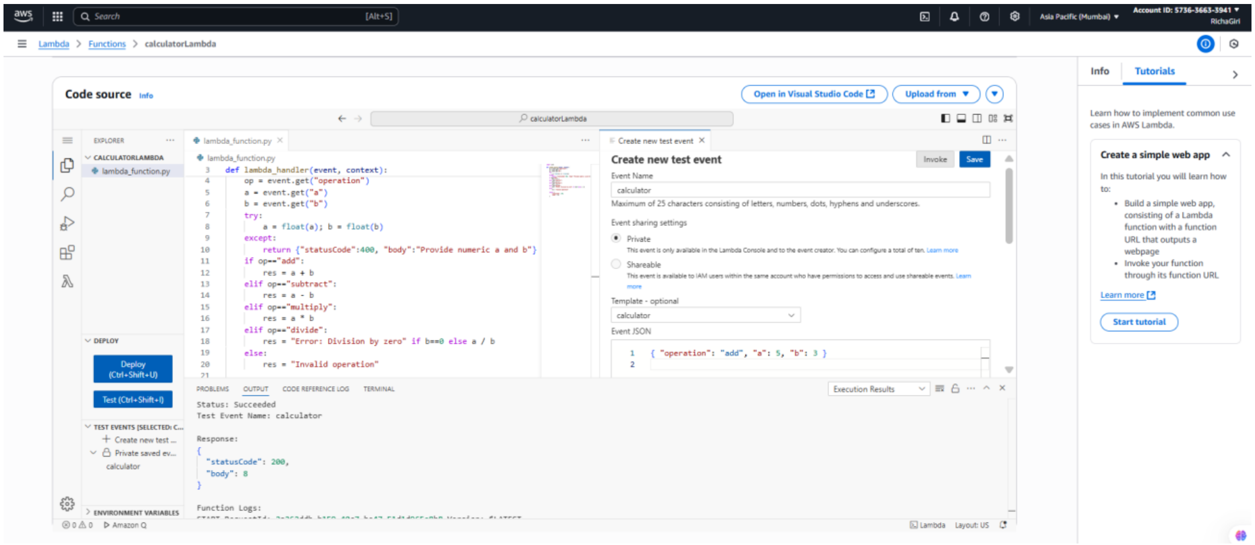
Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more



Code source

```python
def lambda_handler(event, context):
    op = event.get("operation")
    a = event.get("a")
    b = event.get("b")
    try:
        a = float(a); b = float(b)
    except:
        return {"statusCode":400, "body":"Provide numeric a and b"}
    if op=="add":
        res = a + b
    elif op=="subtract":
        res = a - b
    elif op=="multiply":
        res = a * b
    elif op=="divide":
        res = "Error: Division by zero" if b==0 else a / b
    else:
        res = "Invalid operation"
```

Create new test event

Event Name: calculator

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

● Private

This event is only available in the Lambda Console and to the event creator. You can configure a total of ten. Learn more

○ Shareable

This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

Template - optional: calculator

Event JSON

```json
{ "operation": "add", "a": 5, "b": 3 }
```

```
Status: Succeeded
Test Event Name: calculator

Response:
{
  "statusCode": 200,
  "body": 8
}

Function Logs:
```

Learn how to implement common use cases in AWS Lambda.

Create a simple web app

In this tutorial you will learn how to:

- Build a simple web app, consisting of a Lambda function with a function URL that outputs a webpage
- Invoke your function through its function URL

Learn more